CLAIMS

Sub a1

1. A method, comprising:

while executing a program on a computer, detecting the occurrence of profileable events occurring in the instruction pipeline, and directing the instruction pipeline to record profile information describing the profileable events essentially concurrently with the occurrence of the profileable events, the detecting and recording occurring under control of hardware of the computer without software intervention.

2. The method of claim 1, wherein at least one of the recorded profileable events indicates the address of the last byte of an instruction executed by the computer during the profiled execution interval.

3. The method of claim 1, wherein:

at least one of the recorded profileable events notes the source and destination of a control flow event in which control flow of the program execution diverges from sequential execution.

4. The method of claim 1, further comprising the steps of:

executing the program on a first CPU of a multiprocessor computer;

on a second CPU of the multiprocessor, while the execution and profiling of the program continues, analyzing the collected profile data;

controlling the execution of the program on the first CPU based at least in part on the analysis of the collected profile data.

5. The method of claim 1, wherein:

the profile information is recorded into general registers of the computer under hardware control, without software intervention, and without first storing the profile information into main memory.

Sub a2

6. The method of claim 5, further comprising:

when an instruction fetch of an instruction causes a miss in a translation look aside buffer (TLB), the fetch of the instruction triggering a profileable event, servicing the TLB miss and reflecting the corrected state of the TLB in the profile information recorded for the profileable instruction.

Sub B3

7. The method of claim 5:

wherein the program execution induces the occurrence of events in a predefined class of profileable events, with no profile information being recorded in response to the occurrence of profileable events;

the method further comprising:

after a triggering event is detected, the triggering event being one of a predefined class of triggering events, continuing the execution of the program, the continued execution inducing profileable events; and

recording profile entries in a memory of the computer, each profile entry describing one of the profileable events induced after the triggering event.

Sub B4

8. The method of claim 7, further comprising:

recording profile information that records a sequence of events of the program, the sequence including every event during the profiled execution interval following the triggering event that matches time-independent criteria of profileable events to be profiled.

9. The method of claim 7, wherein the triggering event is expiration of a timer.

10. The method of claim 1, wherein the recorded physical memory references include addresses of binary instructions referenced by an instruction pointer, and at least one of the recorded instruction references records the event of a sequential execution flow across a page boundary in the address space.

11. The method of claim 10, wherein at least one of the recorded instruction references records the event of a page boundary of the address space occurring within a single instruction.

12. The method of claim 10, wherein at least one of the recorded instruction references records the event of a page boundary between two instructions that are sequentially adjacent in the logical address space.

Sub B5

13. The method of claim 1, further comprising:

recording profile information recording a processor mode that determines the meaning of binary instructions of the computer.

14. The method of claim 1, further comprising:

recording profile information recording a data-dependent change to a full/empty mask for registers of the computer.

A3

15. The method of claim 1, the recorded profile information being efficiently tailored to identify all bytes of object code executed during the profiled execution interval, without reference to the binary code of the program.

Sub C6

16. The method of claim 1, further comprising:

recording profile information that records a sequence of events of the program, the sequence including every event during the profiled execution interval that matches time-independent criteria of profileable events to be profiled.

17. The method of claim 1, wherein:

the recorded profile information indicates ranges of instruction binary text executed by the computer during a profiled interval of the execution, the ranges of executed text being recorded as low and high boundaries of the respective ranges.

18. The method of claim 1, wherein:

the recorded profile information comprises subunits of two kinds, a first subunit kind describing an instruction interpretation mode at an instruction boundary, and a second subunit kind describing a transition between processor modes.

19. The method of claim 1:

wherein the program execution induces occurrence of events that match time-independent criteria of profileable events to be profiled;

and further comprising, during a profile-quiescent interval of execution, recording no profile information in response to the occurrence of profileable events; and

initiating the profiled execution interval after a triggering event is detected, the triggering event being one of a predefined class of triggering events, the recorded profile information describing the profileable events induced after the triggering event.

20. The method of claim 19, wherein the triggering event is expiration of a timer.

21. The method of claim 19:

wherein the criteria for profileable events divide the profileable events into initiating events and non-initiating events;

and further comprising the steps:

after the triggering event is detected, ignoring non-initiating events; and

when an initiating event is detected, commencing recording the profile entries in the memory, describing every initiating and non-initiating event matching the profileable criteria during an interval following the triggering event.

22. The method of claim 1, further comprising the step of recording a timestamp describing a time of the recorded events.

23. The method of claim 1, wherein at least a portion of the recording is performed by instructions speculatively introduced into the instruction pipeline.

24. Computer hardware, comprising:

an instruction pipeline comprising an arithmetic unit and configured to execute instructions received from a memory of the computer and the profile circuitry; and profile circuitry under common hardware control with the instruction pipeline, the profile circuitry and instruction pipeline cooperatively interconnected to detect the occurrence of profileable events occurring in the instruction pipeline, the profile circuitry operable without software intervention to effect recording of profile information describing the profileable events essentially concurrently with the occurrence of the profileable events.

25. The computer hardware of claim 24:

the profile circuitry being configured to record at least one physical memory reference noting the source and destination of a control flow event in which control flow of the program execution diverges from sequential execution.

26. The computer hardware of claim 24:

wherein the instruction pipeline and profile circuitry are components of a first CPU of a multiprocessor;

and further comprising a second CPU of the multiprocessor, configured to analyze the recorded profile information while the execution and profiling of the program continues on the first CPU, and to at least partially control the operation of the first CPU based at least in part on the analysis of the collected profile data.

27. The computer hardware of claim 24, wherein:

the profile circuitry is configured to record profile information into general registers of the computer under hardware control, without software intervention, and without first storing the profile information into main memory.

28. The computer hardware of claim 27, wherein the recorded profile information is efficiently tailored to identify all bytes of object code executed during the profiled interval, without reference to the binary code of the program.

29. The computer hardware of claim 27, wherein:

the profile circuitry is designed to record profile information that records a sequence of events of the program during the profiled execution interval, the sequence including every event that matches time-independent criteria of profileable events to be profiled.

30. The computer hardware of claim 27, wherein:

the recorded profile information indicates ranges of instruction binary text executed by the computer during a profiled interval of the execution, the ranges of executed text being recorded as low and high boundaries of the respective ranges.

31. The computer hardware of claim 27, wherein:

during a profile-quiescent interval of execution of a program that has been compiled without special consideration for execution profiling and that induces events that match time-independent criteria of profileable events to be profiled, the profile circuitry is configured to record no profile information in response to the occurrence of profileable events; and

after a triggering event is detected, the triggering event being one of a predefined class of triggering events, the profile circuitry is configured to commence the profiled execution interval and to record profile information describing every event that matches the profileable event selection criteria induced during the profiled execution interval, the recording continuing until a predetermined stop condition is reached.

32. The computer hardware of claim 27, further comprising:

a register pointer being a register of the computer, indicating a general register into which to record the profile information; and

an incrementer configured to increment the value of the register pointer to indicate a next general register into which to record next profile information, the incrementing occurring without software intervention.

33. The computer hardware of claim 32, further comprising:

a limit detector operatively interconnected with the register pointer to detect when a range of registers available for collecting profile information is exhausted; and a store unit operatively interconnected with the limit detector of effect storing the profile information from the general registers to the main memory of the computer when exhaustion is detected.

34. The computer hardware of claim 24, further comprising:

profile control bits implemented in the computer hardware, values of the profile control bits controlling a resolution of the operation of the profile circuitry;

the computer having a binary translator configured to translate programs coded in a first instruction set architecture into instructions of a second instruction set architecture, and a profile analyzer configured to analyze the recorded profile information, and to set the profile control bits to values to improve the operation of the binary translator.

35. The computer hardware of claim 24, wherein:

the profile circuitry is interconnected with the instruction pipeline to direct the recording by injection of an instruction into the pipeline, the instruction controlling the pipeline to cause the profileable event to be materialized in an architecturally-visible storage register of the computer.

36. The computer hardware of claim 35, wherein the instruction pipeline and profile circuitry are operatively interconnected to effect speculative injection of the instruction into the instruction pipeline by the profile circuitry.

37. The computer hardware of claim 24, wherein the instruction pipeline and profile circuitry are operatively interconnected to effect injection of multiple instructions into the instruction pipeline by the profile circuitry on the occurrence of a single profileable event.

38. The computer hardware of claim 24 wherein:

an instruction of the computer, having a primary effect on the execution the computer not related to profiling, has an immediate field for an event code encoding the nature of a profiled event and to be recorded in the profile information, the immediate field having no effect on computer execution other than to determine the event code of the profiled event.

39. The computer hardware of claim 38, wherein instances of the instruction have an event code that leaves intact an event code previously determined by other event monitoring circuitry of the computer.

40. The computer hardware of claim 38, wherein the profiled information further includes descriptions of events whose event codes were classified by instruction execution hardware, without any explicit immediate value being recorded in software.

41. The computer hardware of claim 24, wherein:

the profile circuitry comprises a plurality of storage registers arranged in a plurality of pipeline stages, information recorded in a given pipeline stage being subject to modification as a corresponding machine instruction progresses through the instruction pipeline.

42. The computer hardware of claim 24, wherein:

when an instruction fetch of an instruction causes a miss in a translation look aside buffer (TLB), the fetch of the instruction triggering a profileable event, the TLB miss is serviced, and the corrected state of the TLB is reflected in the profile information recorded for the profileable instruction.

43. The computer of claim 24, further comprising profile control bits including a timer interval value specifying a frequency at which the profile circuitry is to monitor the instruction pipeline for profileable events.

44. The computer of claim 24, wherein:

the instruction pipeline is configured to execute instructions of two substantially disjoint instruction sets, a native instruction set providing access to substantially all of the resources of the computer, and a non-native instruction set providing access to a subset of the resources of the computer.

45. The computer of claim 44, wherein the instruction pipeline and profile circuitry are further configured to effect recording of profile information describing an interval of the execution of an operating system coded in the non-native instruction set.